

Granular association rules on two universes with four measures

Fan Min^{a,*}, Qinghua Hu^b, William Zhu^a

^a*Lab of Granular Computing, Zhangzhou Normal University, Zhangzhou 363000, China*

^b*Tianjin University, Tianjin 300072, China*

Abstract

Relational association rules reveal patterns hidden in multiple tables. Existing rules are usually evaluated through two measures, namely support and confidence. However, these two measures may not be enough to describe the strongness of rules. In this paper, we introduce granular association rules with four measures to reveal connections between concepts in two universes, and propose three algorithms for rule mining. Two examples of such associations might be “men like alcohol” and “young men like France alcohol.” With four measures, namely source coverage, target coverage, source confidence and target confidence, our rules are semantically richer than existing ones. Three subtypes of rules are obtained through considering special requirements on the source confidence and the target confidence. Then we define a rule mining problem, and design a sandwich algorithm with different rule checking approaches for different subtypes. Experiments on a real world dataset show that the approaches dedicated to three subtypes are 2-3 orders of magnitudes faster than the one for the general case. Moreover, a forward algorithm and a backward algorithm for one particular subtype can speed up the mining process further. This work opens a new research trend concerning relational association rule mining, granular computing and rough sets.

Keywords: Granular computing, relational association rule, measure, concept, complete match, partial match.

1. Introduction

Relational data mining approaches [10, 11] look for patterns that involve multiple tables in the database. Important issues include relational association rule discovery (see, e.g., [1, 8, 9, 13, 14, 19]), relational decision trees (see, e.g., [6, 22]), and relational distance-based learning (see, e.g., [12]). These issues are

*Corresponding author. Tel.: +86 133 7690 8359

Email addresses: minfanphd@163.com (Fan Min), huqinghua@hit.edu.cn (Qinghua Hu), williamfengzhu@gmail.com (William Zhu)

undoubtedly more general and more challenging than their counterparts on a single data table. Therefore they become popular in recent years.

People have proposed various types of relational association rules for different applications. For example, Dehaspe et al. [9] chained binary relations to produce ternary relations, quaternary relations, etc, and then constructed rules from new relations. Jensen et al. [19] joined a number of primary tables through the central relationship table, then constructed rules from the new table. Goethals et al. [14] constructed rules from two queries, where one asks for a set of tuples satisfying a certain condition, and the other asks for those tuples satisfying a more specific condition. Kavurucu et al. [21] considered one-to-many relationships, and induced logical patterns valid for given background knowledge through Inductive Logic Programming (ILP). Goethals et al. [13] also constructed rules from frequent itemsets across entities and binary relations, with a key specified such that the occurrences of itemsets are counted in one entity table.

These rules are usually evaluated through two measures, namely *support* and *confidence*, which are well defined for association rules [2, 36] in a single data table. Unfortunately, these two measures may not be enough to describe the strongness of relational association rules. For example, according to [13] we may obtain a rule “75% female professors teach courses with 10 credits, among 30% of all courses.” In fact, a professor may teach only one course with 10 credits, or she may teach all courses with 10 credits. Neither measure distinguishes this kind of difference.

In this paper, we introduce granular association rules with four measures to reveal connections between concepts in two universes. The term “granular” comes from granular computing [25, 40, 42, 44], which is an emerging conceptual and computing paradigm of information processing [5]. It indicates that concepts can take any granule specified by an attribute subset. Let us consider a database with two entities **customer** and **product** connected by a relation **buys**. Examples of granular association rules include “men like alcohol,” “young men like France alcohol,” and “Chinese women like white stuff.” The second rule has a *finer* granule than the first one since young men is a subset of men, and France alcohol is a subset of alcohol. However, the third rule neither finer or coarser than the second one because the left part of the second rule is concerned with age and gender, while the left part of the third one is concerned with country and gender. Naturally, a direct application of this type of rules is product recommendation, often referred to as collaborative recommendation [4] or collaborative filtering [15].

We propose four measures to evaluate the quality of a granular association rule. An example of such a rule might be “40% men like at least 30% kinds of alcohol; 45% customers are men and 6% products are alcohol.” Here 45%, 6%, 40%, and 30% are the *source coverage*, the *target coverage*, the *source confidence*, and the *target confidence*, respectively. The *support* measure, which is well defined for other association rules, is redundant since it is equal to the product of the source coverage and the source confidence. With these four measures, the strongness of the rule is well defined. This is one reason why the

new type of rules are semantically richer than most of existing ones. The reason lies in that the new type is more specific than some existing ones which span across more than two universes (see, e.g., [9, 19]) or even the whole database (see, e.g., [13, 14]).

In some cases the source confidence and/or the target confidence might be 100%, resulting in three subtypes with some properties. When the source confidence is 100%, the rule is called a right-hand side partial match one. When the target confidence is 100%, the rule is called a left-hand side partial match one. When both measures are 100%, the rule is called a complete match one. In correspondence with these terms, when neither measure is 100%, the rule is called a partial match one. We may also view partial match rules as a general case without requirements on the source confidence and the target confidence.

Our objective is to mine all granular association rules satisfying thresholds of four measures. We design a sandwich rule mining algorithm for this purpose. With this algorithm, candidate concepts are generated in each universe according to the source coverage and target coverage thresholds using existing algorithms such as Apriori [3] or FP-growth [17]. Then candidate rules are generated and checked. Rules meeting the source confidence and the target confidence thresholds are output. The rule checking approach for partial match rules is inefficient for other subtypes. Therefore we design different rule checking approaches for three subtypes to fully take advantage of their characteristics.

We also design two more algorithms to mine complete matching rules. They are called the forward algorithm and the backward algorithm, respectively. Lower approximation, which is a key concept in rough sets [33], is employed to analyze both algorithms. Hence granular association rule mining can be viewed as as a new application of rough sets.

Experiments are undertaken on the course selection data from Zhangzhou Normal University. Some interesting rules are obtained through setting reasonable thresholds of four measures. The efficiencies of different approaches are compared through different settings on four thresholds. For the sandwich algorithm, rule checking approaches designed for three subtypes are 2-3 orders of magnitude faster than the one for the general case. Moreover, a forward algorithm and a backward algorithm, which are valid for complete matching, can enhance the performance further.

The rest of the paper is organized as follows. Section 2 reviews three types of classical association rules and five types of relational association rules. Section 3 defines the data model for granular association rules and three subtypes of rules. Then Section 4 defines the problem and presents a sandwich algorithm for the problem. A forward algorithm and a backward algorithm are also designed to mine complete match rules. Experiments on the course selection data are discussed in Section 5. Finally, Section 6 presents the concluding remarks and further research directions.

2. Related works

In this section, we review popular association rule mining problems and respective approaches. We will begin with association rules in a single data table, and then proceed to association rules involving multiple tables.

2.1. Association rules

Association rules on a *single* data table have been well-studied. These are boolean association rules, quantitative association rules, and multi-level association rules.

2.1.1. Boolean association rules

The concept of association rule was first introduced in [2] to mine transaction data of a supermarket. This concept was renamed as *boolean association rule* [36] to distinguish with other types of association rules. The transaction data, also called the basket data, store items purchased on a per-transaction basis. An example of such rule is “30% of transactions that contain beer also contain diapers; 2% of all transactions contain both of these items.” Here 30% and 2% are the *confidence* and the *support*, respectively of the rule.

From the set point of view, boolean association rules reveal the connection between two disjoint subsets of the same universe. Let the number of transactions be n and the number of items be m , the basket data can be stored in an information table with n rows and m columns. Each datum in the data table is boolean to specify whether or not an item is included in the respective transaction. This is why the rules are called boolean association rules.

The Apriori [2, 3] algorithm is based on the Apriori property [3]. It can mine all boolean association rules efficiently given the threshold of support and confidence. The FP-growth [17] algorithm avoids candidate generation and therefore save computation time further.

2.1.2. Quantitative association rules

Quantitative association rule [36] was introduced to cope with data tables with quantitative attribute values. From the data type point of view, it is a generalization of the boolean association rule. It reveals the relationships among attribute values of an object. A well known application is mining information of people. An example of such rule is “10% of married people between age 50 and 60 have at least 2 cars; 3% of all people queried satisfy this rule” [36]. Similar to the case of boolean association rules, here 10% is called the *confidence* of the rule, and 3% is the *support* of the rule.

Since the Apriori property still holds in the new context, the Apriori algorithm can be designed accordingly [36]. One can also follow the idea of FP-growth to design a more efficient algorithm.

2.1.3. Multi-level association rules

Multi-level association rules [16] reside at multiple concept levels to discover more specific and concrete knowledge from data. In addition to the transaction data, it requires a description table to indicate different levels. Suppose that category, content and brand represent the first, the second, and the third level concept respectively of a food. Two examples of such rules are “75% of people buy wheat bread if they buy 2% milk,” and “82% of people buy bread if they buy 2% milk.” However, the rule “60% of people buy products made of wheat if they buy 2% milk” is invalid since “products made of wheat” does not indicate the category.

2.2. Relational association rules

In recent years, multi-relational data mining (MRDM) [10], also called relational data mining (RDM), has been proposed to look for patterns that involve multiple tables. Accordingly, the concept of association rule has been extended with this regard to form relational association rules. There are various extensions, and we will discuss more popular ones.

2.2.1. Extended boolean association rules

Dehaspe et al. [9, 8], Džeroski et al. [10, 11], and Afrati et al. [1] considered the case where binary relations can be chained to produce ternary relations, quaternary relations, etc. Suppose there are two binary relations, namely the parent-child relation and child-pet relation. A parent-child-pet relation can be produced using a SQL query on the database. An example of such rule is “if a person has a child, then this child has a pet with a probability of 30%; 20% of all people satisfy this rule.” Here 30% is called the *confidence* of the rule, and 20% is the *support* of the rule.

We will call this type of rules *extended boolean association rules* since they can be viewed a direct extension of boolean association rules on a single table. The information carried by such rules are quite limited. They cannot indicate the number of children a person has, or the number of pets a child has. Nor can they specify other information, such as the age, of a parent or a child.

Dehaspe et al. [8] designed a general purpose inductive logic programming algorithm called WARMR to mine this type of rules. Afrati et al. [1] also tried to attack this problem using integer programming and graph approaches.

2.2.2. Decentralized association rules

Jensen et al. [19] considered the case of decentralize tables. In this case the database contains n primary tables (i.e., tables with one primary key), and one central relationship table (i.e., a table with n foreign keys). An example of such rule is “if the ATM type is drive, then the age of the customer is between 20 and 29.” The computation of the confidence and support measures is the same as the table joined from all $n + 1$ tables.

We will call this type of rules *decentralized association rules*. In fact, if $n = 2$, the database represents a many-to-many relation, which is quite typical.

However, in real applications a central relationship table seldom exists for $n > 2$. Therefore these rules are valid for very special databases, or parts of a database.

2.2.3. Simple conjunctive association rules

Goethals et al. [14] considered mining association rules in arbitrary relational databases. This approach looks for pairs of SQL queries Q_1 and Q_2 , such that “ Q_1 asks for a set of tuples satisfying a certain condition and Q_2 asks for those tuples satisfying a more specific condition” [14]. When the number of tuples matching Q_2 is close to that of Q_1 , a rule is created. An example of such rule is “actors starring in ‘drama’ movies typically (with a probability of 90%) also star in a ‘comedy’ movie.”

We will call this type of rules *simple conjunctive association rules*. The conjunction here is much more flexible than the case of extended boolean association rules. In fact, any kind of SQL query is supported. Goethals et al. [14] designed the Conqueror algorithm to mine this type of rules.

2.2.4. IPL-based association rules

Kavurucu et al. [21] considered one-to-many relationships. They extended the background knowledge with aggregate predicates in order to characterize the structural information that is stored in tables and association between them. In this way, logical patterns valid for given background knowledge are induced through Inductive Logic Programming (ILP)

We will call this type of rules *IPL-based association rules*. Kavurucu et al. [21] designed a concept discovery system named Confidence-based Concept Discovery (C²D). C²D does not require user specification of input/output modes of arguments. Therefore it is suitable for non-expert users without much knowledge on the semantic detail of the relations.

2.2.5. Separated counting association rules

Goethals et al. [13] also considered a more specific type of association rules. The frequency of a rule is not counted as the number of occurrences in the join of tables. Let the database consist tables **Professor**, **Course** and **Student**. For one particular kind of courses, the number of professors who teach them and the number of students who study them are counted separately. An example of such rule is “75% professors named Jan teaches courses with 10 credits, among 30% of all courses.” Here 75% is the *confidence* and 30% is the *relative support*.

We will call this type of rules *separated counting association rules*. Unfortunately, the counting mechanism is not good enough. For example, a professor may teach only one course with 10 credits, or she may teach all courses with 10 credits. This type of rules does not contain such information.

Unfortunately, there are at least two drawbacks of existing relational association rule mining works. First, the general relational association rule mining problem usually involves the join operation of multiple data tables [19]. When the sizes of these data tables are large, it is simply impossible to join more than two tables. Second, as the association rule becomes more complex in the context

Table 1: Customer

CID	Name	Age	Gender	Married	Country	Income	NumCars
c1	Ron	20..29	Male	No	USA	60k..69k	0..1
c2	Michelle	20..29	Female	Yes	USA	80k..89k	0..1
c3	Shun	20..29	Male	No	China	40k..49k	0..1
c4	Yamago	30..39	Female	Yes	Japan	80k..89k	2
c5	Wang	30..39	Male	Yes	China	90k..99k	2

of RDM, the support and confidence measures are not enough to evaluate the strongness of the rule.

3. Granular association rules with three subtypes

In this section, we will introduce granular association rules to address the drawbacks of existing types mentioned in the last section. We will first discuss the data model for the new type. Then we present three subtypes of rules and one general case corresponding to four different explanations of granular association rules. At the same time, a number of measures are proposed to evaluate the quality of these rules. A comprehensive comparison with existing types will be made at the end of the section.

3.1. The data model

First we need to revisit the definitions of information systems and binary relations.

Definition 1. $S = (U, A)$ is an information system, where $U = \{x_1, x_2, \dots, x_n\}$ is the set of all objects, $A = \{a_1, a_2, \dots, a_m\}$ is the set of all attributes, and $a_j(x_i)$ is the value of x_i on attribute a_j for $i \in [1..n]$ and $j \in [1..m]$.

An example of information system is given by Table 1, where $U = \{c1, c2, c3, c4, c5\}$, and $A = \{\text{Age, Gender, Married, Country, Income, NumCars}\}$. Another example is given by Table 2.

In an information system, any $A' \subseteq A$ induces an equivalent relation [33, 35]

$$E_{A'} = \{(x, y) \in U \times U \mid \forall a \in A', a(x) = a(y)\}, \quad (1)$$

and partitions U into a number of disjoint subsets called *blocks*. The block containing $x \in U$ is

$$E_{A'}(x) = \{y \in U \mid \forall a \in A', a(y) = a(x)\}. \quad (2)$$

From another viewpoint, a pair $C = (A', x)$ where $x \in U$ is called a *concept*. The *extension* of the concept is

$$ET(C) = ET(A', x) = E_{A'}(x); \quad (3)$$

Table 2: Product

PID	Name	Country	Category	Color	Price
p1	Bread	Australia	Staple	Black	1..9
p2	Diaper	China	Daily	White	1..9
p3	Pork	China	Meat	Red	1..9
p4	Beef	Australia	Meat	Red	10..19
p5	Beer	France	Alcohol	Black	10..19
p6	Wine	France	Alcohol	White	10..19

while the *intension* of the concept is the conjunction of respective attribute-value pairs, i.e.,

$$IT(C) = IT(A', x) = \bigwedge_{a \in A'} \langle a : a(x) \rangle. \quad (4)$$

The *support* of the concept is the size of its extension divided by the size of the universe, namely,

$$\begin{aligned} support(C) &= support(A', x) = support(\bigwedge_{a \in A'} \langle a : a(x) \rangle) \\ &= support(E_{A'}(x)) = \frac{|ET(A', x)|}{|U|} \\ &= \frac{|E_{A'}(x)|}{|U|}. \end{aligned} \quad (5)$$

Definition 2. Let $U = \{x_1, x_2, \dots, x_n\}$ and $V = \{y_1, y_2, \dots, y_k\}$ be two sets of objects. $R \subseteq U \times V$ is a binary relation from U to V .

$$R(x) = \{y \in V \mid (x, y) \in R\}, \quad (6)$$

$$R^{-1}(y) = \{x \in U \mid (x, y) \in R\}. \quad (7)$$

A binary relation is more often stored in the database as a table with two foreign keys. In this way the storage is saved. For the convenience of illustration, here we represented it with an $n \times k$ boolean matrix. An example is given by Table 3, where U is the set of customers as indicated by Table 1, and V is the set of products as indicated by Table 2.

With Definitions 1 and 2, we propose the following definition.

Definition 3. A many-to-many entity-relationship system (MMER) is a 5-tuple $ES = (U, A, V, B, R)$, where (U, A) and (V, B) are two information systems, and $R \subseteq U \times V$ is a binary relation from U to V .

An example of MMER is given by Tables 1, 2 and 3.

3.2. Granular association rule with three subtypes

A *granular association rule* is an implication of the form

$$(GR) : \bigwedge_{a \in A'} \langle a : a(x) \rangle \Rightarrow \bigwedge_{b \in B'} \langle b : b(y) \rangle, \quad (8)$$

Table 3: Buys

CID \ PID	p1	p2	p3	p4	p5	p6
c1	1	1	0	1	1	0
c2	1	0	0	1	0	1
c3	0	1	1	0	1	1
c4	0	1	0	1	1	0
c5	1	0	1	1	1	1

where $A' \subseteq A$ and $B' \subseteq B$.

According to Equation (5), the set of objects meeting the left-hand side of the granular association rule is

$$LH(GR) = E_{A'}(x); \quad (9)$$

while the set of objects meeting the right-hand side of the granular association rule is

$$RH(GR) = E_{B'}(y). \quad (10)$$

We define two measures to evaluate the generality of the granular association rule. The *source coverage* of GR is

$$scoverage(GR) = \frac{|LH(GR)|}{|U|}; \quad (11)$$

while the *target coverage* of GR is

$$tcoverage(GR) = \frac{|RH(GR)|}{|V|}. \quad (12)$$

In most cases, rules with higher source coverage and target coverage tend to be more interesting. We present a granular association rule for discussion.

$$\begin{aligned} \langle \text{Gender: Male} \rangle &\Rightarrow \langle \text{Category: Alcohol} \rangle \\ [scoverage = 60\%, tcoverage = 33\%]. \end{aligned} \quad (13)$$

A direct explanation of Rule (13) is “men like alcohol.” However, this explanation is ambiguous and the following questions may arise: Do all men like alcohol? Do men like all kinds of alcohol? To avoid such ambiguity, more measures of the rule are needed. We propose four different explanations of this rule, as illustrated in Figure 1, and will discuss them from simple ones to more general ones. Note that exemplary rules discussed in the following context may not comply to the MMER given by Tables 1, 2 and 3.

3.2.1. Complete match

The first explanation of Rule (13) is “all men like all alcohol,” or equivalently, “100% men like 100% alcohol.” This can be formally expressed by the following definition.

Complete match rule: “All men like all kinds of alcohol.”		<div>special</div> <div>↑</div> <div>↓</div> <div>general</div>
Left-hand side partial match rule: “40% men like all kinds of alcohol.”	Right-hand side partial match rule: “All men like at least 30% kinds of alcohol.”	
Partial match rule: “40% men like at least 30% kinds of alcohol”		

Figure 1: Four explanations of “men like alcohol”

Definition 4. A granular association rule GR is called a *complete match granular association rule* iff

$$LH(GR) \times RH(GR) \subseteq R. \quad (14)$$

It is also called a *complete match rule* for brevity. We need to know the percentage of objects in U matching the rule. It is called the *support* of the rule and defined by

$$support_c(GR) = scoverage(GR) = \frac{|LH(GR)|}{|U|}, \quad (15)$$

where the suffix c stands for *complete*. Although the support is equal to the source coverage, we still define this measure since in other subtypes they are different. Under this context, the rule

$$\begin{aligned} \langle \text{Gender: Male} \rangle &\Rightarrow \langle \text{Category: Alcohol} \rangle \\ [scoverage = 60\%, tcoverage = 33\%], \end{aligned} \quad (16)$$

will be read as “all men like all kinds of alcohol; 60% of all people are men; 33% of all products are alcohol.” Note that Rules (13) and (16) have the same form. However the explanation of Rule (16) causes no ambiguity under the context of complete match.

3.2.2. Left-hand side partial match

The second explanation of Rule (13) is “some men like all alcohol,” or equivalently, “at least one man like 100% alcohol.” Because “some” appears on the left-hand side, the rule is called “left-hand side partial match.” Consequently, we define a subtype of granular association rule as follows.

Definition 5. A granular association rule GR is called a *left-hand side partial match rule* iff there exists $x \in LH(GR)$ such that

$$R(x) \supseteq RH(GR). \quad (17)$$

In applications, however, if very few men like all kinds of alcohol, this rule is not quite useful. We need to know the percentage of men that like alcohol. The *support* of the rule is

$$support_{lp}(GR) = \frac{|\{x \in LH(GR) | R(x) \supseteq RH(GR)\}|}{|U|}. \quad (18)$$

In other words, only men that like all kinds of alcohol are counted. Moreover, the *source confidence* of the rule is

$$sconfidence_{lp}(GR) = \frac{|\{x \in LH(GR) | R(x) \supseteq RH(GR)\}|}{|LH(GR)|}. \quad (19)$$

One may obtain the following rule

$$\begin{aligned} &\langle \text{Gender: Male} \rangle \Rightarrow \langle \text{Category: Alcohol} \rangle \\ &[scoverage = 60\%, tcoverage = 33\%, sconfidence_{lp} = 67\%], \end{aligned} \quad (20)$$

which is read as “67% men like all kinds of alcohol; 60% of customers are men; 33% of products are alcohol.” We deliberately avoid the support measure in this explanation; the reason will be discussed in the next subsection.

3.2.3. Right-hand side partial match

The third explanation of Rule (13) is “all men like some kinds of alcohol,” or equivalently, “100 % men like at least one kind of alcohol.” Because “some” appears on the right-hand side, the rule is called “right-hand side partial match.” Consequently, we define a subtype of granular association rule as follows.

Definition 6. A granular association rule GR is called a *right-hand side partial match rule* iff $\forall x \in LH(GR)$,

$$R(x) \cap RH(GR) \neq \emptyset. \quad (21)$$

Similar to the case of complete match, the *support* of the rule is equal to the source coverage. It is given by

$$support_{rp}(GR) = scoverage(GR) = \frac{|LH(GR)|}{|U|}. \quad (22)$$

In the case of complete match and left-hand side partial match, bigger target coverage values indicate stronger rules. Unfortunately, in the case of right-hand side partial match, bigger target coverage values indicate weaker rules. Consider one extreme case as follows: “all customers like at least one kind of all products.” The rule always holds, and both the source coverage and the target coverage of the rule are 100%, but the rule is totally useless.

Therefore we need to know how many kinds of alcohol men like. Here we introduce a new measure called *target confidence* for this purpose. The *target confidence* of the right-hand side partial match rule is

$$tconfidence_{rp}(GR) = \min_{x \in LH(GR)} \frac{|R(x) \cap RH(GR)|}{|RH(GR)|}. \quad (23)$$

With existing measures, we may obtain the following rule

$$\langle \text{Gender: Male} \rangle \Rightarrow \langle \text{Category: Alcohol} \rangle$$

$$[scoverage_{rp} = 60\%, tcoverage = 33\%, tconfidence_{rp} = 50\%]. \quad (24)$$

which is read as “all men like at least 50% of alcohol; 60% of customers are men; 33% of products are alcohol.”

3.2.4. Partial match

The fourth explanation of Rule (13) is “some men like some kinds of alcohol,” or equivalently, “at least one man like at least one kind of alcohol.” Because “some” appears on both sides, the rule will be simply called “partial match.” Consequently, we define this type of granular association rule as follows.

Definition 7. A granular association rule GR is called a *partial match granular association rule* iff there exists $x \in LH(GR)$ and $y \in RH(GR)$ such that

$$(x, y) \in R. \quad (25)$$

It is also called a *partial match rule* for brevity. According to the definition, partial match is a general case of granular association rules. Therefore we cannot call it a *subtype*.

There is a tradeoff between the source confidence and the target confidence of a rule. Consequently, neither value can be obtained directly from the rule. To compute any one of them, we need to specify the threshold of the other. Let tc be the target confidence threshold. The *support* of the partial match rule is

$$support_p(GR, tc) = \frac{|\{x \in LH(GR) \mid \frac{|R(x) \cap RH(GR)|}{|RH(GR)|} \geq tc\}|}{|U|}. \quad (26)$$

Here tc is a necessary parameter. For convenience, in some cases we may ignore it to keep the same form as others. The *source confidence* of the partial match rule is

$$sconfidence_p(GR, tc) = \frac{|\{x \in LH(GR) \mid \frac{|R(x) \cap RH(GR)|}{|RH(GR)|} \geq tc\}|}{|LH(GR)|}. \quad (27)$$

Let mc be the source confidence threshold, and

$$\begin{aligned} & |\{x \in LH(GR) \mid |R(x) \cap RH(GR)| \geq K + 1\}| \\ & < mc \times |LH(GR)| \\ & \leq |\{x \in LH(GR) \mid |R(x) \cap RH(GR)| \geq K\}|. \end{aligned} \quad (28)$$

The *target confidence* of the partial match rule is

$$tconfidence_p(GR, mc) = \frac{K}{|RH(GR)|}. \quad (29)$$

In fact, the computation of K is non-trivial. First, for any $x \in LH(GR)$, we need to compute $tc(x) = |R(x) \cap RH(GR)|$ and obtain an array of integers. Second,

Table 4: Summary of source confidence and target confidence

Subtype \ Measure	Source confidence	Target confidence
Complete match	100%	100%
Left-hand side partial match	$\frac{ \{x \in LH(GR) R(x) \supseteq RH(GR)\} }{ LH(GR) }$	100%
Right-hand side partial match	100%	$\min_{x \in LH(GR)} \frac{ R(x) \cap RH(GR) }{ RH(GR) }$
Partial match	$\frac{ \{x \in LH(GR) \frac{ R(x) \cap RH(GR) }{ RH(GR) } \geq tc\} }{ LH(GR) }$	$\frac{K}{ RH(GR) }$

we sort the array in a descending order. Third, let $k = \lfloor mc \times |LH(GR)| \rfloor$, K is the k -th element in the array.

With existing measures, we may obtain the following rule

$$\begin{aligned} &\langle \text{Gender: Male} \rangle \Rightarrow \langle \text{Category: Alcohol} \rangle \\ &[scoverage = 60\%, tcoverage = 33\%, sconfdence_p = 40\%, tconfidence_p = 30\%]. \end{aligned} \quad (30)$$

which is read as “40% men like at least 30% of alcohol; 60% of customers are men; 33% of products are alcohol.”

3.3. Discussion of measures

We have presented five measures to evaluate the quality of granular association rules. The source coverage is always $\frac{|LH(GR)|}{|U|}$, and the target coverage is always $\frac{|RH(GR)|}{|U|}$. Table 4 summaries source confidence and target confidence. From Equations (15), (18), (19), (22), (26) and (27) we know that for all four cases, there is a direct connection among the support, source coverage and confidence of a rule.

$$support_*(GR) = scoverage(GR) \times sconfdence_*(GR), \quad (31)$$

where the suffix “*” could be replaced by c , lp , rp and p . Hence any one of these three measures can be viewed redundant. For convenience, in the following context we will ignore the support measure.

3.4. Alternative definitions

It is worth noting that Definitions 5 and 6 are asymmetric. A symmetric definition of Definition 5 is

Definition 8. A granular association rule GR is called a *type-2 right-hand side partial match rule* iff there exists $y \in RH(GR)$ such that

$$R^{-1}(y) \supseteq LH(GR). \quad (32)$$

With Definition 8, we have the following explanation of the rule “at least one kind of alcohol favors all men.” Moreover, a symmetric definition of Definition 6 is

Definition 9. A granular association rule GR is called a *type-2 left-hand side partial match rule* iff $\forall y \in RH(GR)$

$$R^{-1}(y) \cap LH(GR) \neq \emptyset. \quad (33)$$

With Definition 9, we have the following explanation of the rule “all kinds of alcohol favors at least one man.” Unfortunately, the subject these new rules are concepts in V , and the relation under consideration is R^{-1} instead of R . Therefore the alternative definitions are not appropriate for our situation.

3.5. Comparison with existing types

We now compare granular association rules with other types of association rules mentioned in Section 2.

1. Both boolean association rules and granular association rules deal with binary relations on two universes. For granular association rules, objects in either universe are described by a number of attributes. Therefore granular association rules reveal connections between object subsets in two universes, while boolean association rules reveal connections between objects in one universe.
2. Both quantitative association rules and granular association rules deal with quantitative data. Moreover, the data sources are all described by attributes. Quantitative association rules involve only one universe, while granular association rules always involve two.
3. Both multi-layer association rules and granular association rules describe objects with attributes. Multi-layer association rules have a predefined concept hierarchy with a tree structure, which does not exist for granular association rules. Moreover, Multi-layer association rules involve only one universe.
4. Extended boolean association rules may involve more than two data tables. Similar to boolean association rules, objects are not described by attributes. Therefore they reveal connections between objects in different universes.
5. Decentralized association rules involve at least two primary tables. From this viewpoint, they are more general than granular association rules. As mentioned earlier, this type of rules have a special requirement on the database. Hence they are less useful than granular association rules.
6. Simple conjunctive association rules are quite flexible. They reveal the connections between a object set and one of its subsets. And the motivation is totally different from granular association rules.
7. IPL-based association rules consider one-to-many relationships, while granular association rules consider many-to-many relationships. Moreover, C²D for IPL-based association rules does not require much user specification, while granular association rules require four thresholds for measures.

8. Granular association rules have the same form as separated counting association rules. The number of objects for a rule is counted locally in one universe, therefore the joining of tables is unnecessary. One important difference between two types lies in that granular association rules have more measures, therefore they are semantically richer.

There is still another closely related technique called collaborative recommendation [4] or collaborative filtering [15]. This technique also considers many-to-many relationships with some interesting applications such as product recommending and web page recommending. Compared with granular association rules, this technique focuses more on particular applications. Therefore from one viewpoint granular association rules can be employed in collaborative recommendation. From another viewpoint, collaborative recommendation can be viewed as a local approach since we always recommend something to a user. In contrast, granular association rules can be viewed as a global approach which outputs only strong rules.

4. Granular association rule mining algorithms

In this section, we first define the granular association rule mining problem. Then we propose a sandwich algorithm with four rule checking approaches, one for partial matching rules and three for subtypes. Naturally, the one for partial matching rules is also valid for three subtypes. Then two more algorithms are designed for the complete match subtype. Time complexities of all algorithms are analyzed.

4.1. The granular association rule mining problem

We now define the problem as follows.

Problem 10. *The granular association rule mining problem.*

Input: An $ES = (U, A, V, B, R)$, a minimal source coverage threshold ms , a minimal target coverage threshold mt , a minimal source confidence threshold mc , and a minimal target confidence threshold tc .

Output: All granular association rules satisfying $scoverage(GR) \geq ms$, $tcoverage(GR) \geq mt$, $sconfidence_p(GR) \geq mc$, and $tconfidence_p(GR) \geq tc$.

4.2. A sandwich algorithm

A straightforward algorithm for Problem 10 is given by Algorithm 1. It essentially has three steps.

Step 1. Search in (U, A) all concepts meeting the minimal source coverage threshold ms . This step corresponds to Line 1 of the algorithm, where SC stands for source concept.

Step 2. Search in (V, B) all concepts meeting the minimal target coverage threshold mt . This step corresponds to Line 2 of the algorithm, where TC stands for target concept.

Algorithm 1 A sandwich algorithm for partial match

Input: $ES = (U, A, V, B, R)$, ms , mt , mc , tc .

Output: All partial match rules satisfying given constraints.

Method: partial-match-sandwich

```

1:  $SC(ms) = \{(A', x) \in 2^A \times U \mid \frac{|E_{A'}(x)|}{|U|} \geq ms\};$ 
2:  $TC(mt) = \{(B', y) \in 2^B \times V \mid \frac{|E_{B'}(y)|}{|V|} \geq mt\};$ 
3: for each  $C \in SC(ms)$  do
4:   for each  $C' \in TC(mt)$  do
5:      $GR = (IT(C) \Rightarrow IT(C'));$ 
6:     if  $sconfidence_p(GR, tc) \geq mc$  then
7:       output rule  $GR$ ;
8:     end if
9:   end for
10: end for
```

Step 3. Check all possible rule regarding SC and TC , and output valid ones. This step corresponds to Lines 3 through 10 of the algorithm.

Since this algorithm starts from both ends of the association rule and proceeds to the middle, it is called the “sandwich” algorithm. Note that the check of the condition $sconfidence_p(GR, tc) \geq mc$ in Line 6 is non-trivial. And it indicates both thresholds of source confidence and target confidence should be met.

Now we discuss the algorithm in more detail. The Apriori algorithm [3, 36] and the FP-growth algorithm [17] can be employed in Lines 1 and 2. These algorithms are based on the Apriori property, which is stated as “every subset of a frequent itemset must also be a frequent itemset” [3]. Under our context, the Apriori property can be restated as follows.

Property 11. Let $A'' \subset A' \subseteq A$ and $x \in U$.

$$|E_{A'}(x)| \leq |E_{A''}(x)|. \quad (34)$$

Naturally, for three subtypes, the condition expressed by Line 6 of the algorithm might be replaced by simpler ones. We will explain the cases for each subtypes.

4.2.1. Complete match

If $mc = tc = 100\%$, we are essentially looking for complete match rules. The condition can be replaced by

$$ET(C) \times ET(C') \subseteq R. \quad (35)$$

Moreover, in this case some checks are redundant. We have the following property.

Property 12. Let $A'' \subset A' \subseteq A$, $x \in U$, $B'' \subset B' \subseteq B$, and $y \in V$. If $ET(A'', x) \times ET(B'', y) \subseteq R$,

$$ET(A', x) \times ET(B', y) \subseteq R. \quad (36)$$

PROOF. Because $A'' \subset A'$, $ET(A', x) \subseteq ET(A'', x)$. Similarly $ET(B', y) \subseteq ET(B'', y)$. Therefore $ET(A', x) \times ET(B', y) \subseteq ET(A'', x) \times ET(B'', y)$. And the property holds.

Property 12 is essentially another form of the Apriori property. Its converse negative proposition can be used to remove unnecessary check of rules. Note that changes can be made on both sides of the rule. For example, if rule “all Chinese men like all kinds of France alcohol” does not hold, then rule “all men like all kinds of alcohol” never holds.

4.2.2. Left-hand side partial match

If $tc = 100\%$, we are essentially looking for left-hand side partial match rules. The condition can be replaced by

$$\frac{|\{x \in LH(GR) | R(x) \supseteq RH(GR)\}|}{|LH(GR)|} \geq mc. \quad (37)$$

4.2.3. Right-hand side partial match

If $mc = 100\%$, we are essentially looking for right-hand side partial match rules. The condition can be replaced by

$$\min_{x \in ET(C)} \frac{|R(x) \cap ET(C')|}{|ET(C')|} \geq tc. \quad (38)$$

Similar to the case of complete match, we would like to remove unnecessary check of rules. In fact, we have the following property.

Property 13. Let $A'' \subset A' \subseteq A$, $x \in U$, $B' \subseteq B$, and $y \in V$.

$$\min_{x' \in ET(A', x)} \frac{|R(x) \cap ET(B', y)|}{|ET(B', y)|} \geq \min_{x' \in ET(A'', x)} \frac{|R(x) \cap ET(B', y)|}{|ET(B', y)|}. \quad (39)$$

PROOF. Because $A'' \subseteq A'$, $ET(A'', x) \supseteq ET(A', x)$. Hence Equation (39) holds.

Property 13 indicates one approach to removing unnecessary check concerning the left side of the rule. Unlike Property 12, in this case the change cannot be made on both sides. For example, if rule “all Chinese men like at least 30% kinds of France alcohol” does not hold, then “all men like at least 30% kinds of France alcohol” never holds. However, “all Chinese men like at least 30% kinds of alcohol” may hold.

Now we analyze the time complexity of the algorithm. For the partial match subtype, from Equation (26) we know the complexity of Line 6 is

$$O(|ET(C)| \times |ET(C')|) = O(|U| \times |V|). \quad (40)$$

Algorithm 2 A forward algorithm

Input: $ES = (U, A, V, B, R)$, ms , mt .

Output: All complete match granular association rules satisfying given constraints.

Method: complete-match-rules-forward

```
1:  $SC(ms) = \{(A', x) \in 2^A \times U \mid \frac{|E_{A'}(x)|}{|U|} \geq ms\};$ 
2:  $TC(mt) = \{(B', y) \in 2^B \times V \mid \frac{|E_{B'}(y)|}{|V|} \geq mt\};$ 
3: for each  $C \in SC(ms)$  do
4:    $X = ET(C);$ 
5:    $Y = \underline{R}(X);$ 
6:   for each  $C' \in TC(mt)$  do
7:     if  $(ET(C') \subseteq Y)$  then
8:       output rule  $IT(C) \Rightarrow IT(C')$ ;
9:     end if
10:  end for
11: end for
```

According to the **for** loops, the time complexity of Algorithm 1 is

$$O(|SC(ms)| \times |TC(mt)| \times |U| \times |V|). \quad (41)$$

For the complete match subtype, suppose that both $ET(C)$ and $ET(C')$ are stored in 1-dimensional positive number arrays. Each element in the array indicates the inclusion of one particular object in the concept. For example, $[1, 4, 8]$ indicates $\{x_1, x_4, x_8\}$. Suppose further that R is stored in a $|U| \times |V|$ boolean array. The time complexity of checking $ET(C) \times ET(C') \subseteq R$ is the same as that of partial match as indicated by Equation (40). Consequently, this time complexity for the complete match subtype is also given by Equation (41). However, checking $ET(C) \times ET(C') \subseteq R$ ends immediately once a violation of the relationship is found. Compared with the check of $sconfidence_p(GR, tc) \geq mc$, it is less time consuming.

Similarly, for the other two subtypes, the time complexities are all given by Equation (41). The run time for different subtypes will, however, be very different in applications. This will be shown through experiments in Section 5.

4.3. Two algorithms for the complete match subtype

The time complexity of the sandwich algorithm is quite high. Now we propose two alternative approaches for the complete match subtype. We will show that their time complexities are lower than Algorithm 1.

4.3.1. A forward algorithm

The first alternative approach is called the “forward” approach. It starts from the left-hand side of the rule and proceeds to the right-hand side. The algorithm is listed in Algorithm 2. It essentially has four steps.

Steps 1 and 2. They are the same as Algorithm 1.

Step 3. For each concept obtained in Step 1, construct a block in V according to R . This step corresponds to Line 4 of the algorithm. The function ET has been defined in Equation (3). We introduce a new concept regarding Line 5.

Definition 14. Let U and V be two universes, $R \subseteq U \times V$ be a binary relation, $X \subseteq U$. The lower approximation of X with respect to R is

$$\underline{R}(X) = \{y \in V | R^{-1}(y) \supseteq X\}. \quad (42)$$

In our example, $\underline{R}(X)$ are all products that favor all people in X . The concept “lower approximation” comes from rough sets [33]. However, we consider two universes here instead of only one.

Step 4. Check possible rules regarding C' and Y , and output all rules. This step corresponds to Lines 6 through 10 of the algorithm. In Line 7, since $ET(C')$ and Y could be stored in sorted arrays, the complexity of checking $ET(C') \subseteq Y$ is

$$O(|ET(C')| + |Y|) = O(|V|). \quad (43)$$

According to the **for** loops, the time complexity of Algorithm 2 is

$$O(|SC(ms)| \times |TC(mt)| \times |V|), \quad (44)$$

which is lower than Algorithm 1.

4.4. The backward algorithm

The backward algorithm, which is a dual of Algorithm 2, is listed in Algorithm 3. It starts from the right-hand side of the rule and proceeds to the left-hand side. It is symmetric with respect to Algorithm 2. According to Definition 14, $\underline{R}^{-1}(Y) = \{x \in U | R(x) \supseteq Y\}$. In our example, $\underline{R}^{-1}(Y)$ are all people buying all products in Y . Similar to the analysis of Algorithm 2, the time complexity of Algorithm 3 is

$$O(|SC(ms)| \times |TC(mt)| \times |U|). \quad (45)$$

Now one question arises: which algorithm performs better? According to Equations (44) and (45), we should choose the forward algorithm if $|U| > |V|$, and the backward algorithm otherwise. This issue will be discussed further through experimentation in Section 5.

5. Experiments on a real world dataset

In this section, we try to answer the following problems through experimentation.

1. Do granular association rules make sense in real-world applications?
2. Do dedicated approaches for different subtypes improve the performance of the sandwich algorithm?
3. Do the forward and backward algorithms outperform the sandwich algorithm significantly?

Algorithm 3 The backward algorithm

Input: $ES = (U, A, V, B, R)$, ms , mt .

Output: All complete match granular association rules satisfying given constraints.

Method: complete-match-rules-backward

```

1:  $SC(ms) = \{(A', x) \in 2^A \times U \mid \frac{|E_{A'}(x)|}{|U|} \geq ms\}$ ;
2:  $TC(mt) = \{(B', y) \in 2^B \times V \mid \frac{|E_{B'}(y)|}{|V|} \geq mt\}$ ;
3: for each  $C' \in TC(mt)$  do
4:    $Y = ET(C')$ ;
5:    $X = \underline{R}^{-1}(Y)$ ;
6:   for each  $C \in SC(ms)$  do
7:     if  $(ET(C) \subseteq X)$  then
8:       output rule  $IT(C) \Rightarrow IT(C')$ ;
9:     end if
10:  end for
11: end for

```

5.1. Dataset

We obtained a real-world dataset from Zhangzhou Normal University. The database schema is as follows.

- Student (studentID, name, gender, birth-year, politics-status, grade, department, nationality, length-of-schooling)
- Course (courseID, credit, class-hours, availability, department)
- Selects (studentID, courseID)

We collected data during the semester between 2011 and 2012. There are 145 general education courses in the university, and 9,654 students took part in course selection.

5.2. Results

We undertake three sets of experiments to answer the questions raised at the beginning of the section one by one.

5.2.1. The meaningfulness of rules

We obtain some strong rules using the sandwich algorithm. Let $ms = 0.06$, $mt = 0.06$, $mc = 0.18$, and $tc = 0.11$. 40 granular association rules are obtained, and 4 of them listed below.

(Rule 1) $\langle \text{department: economics} \rangle$

$\Rightarrow \langle \text{department: human-resource} \rangle$

(Rule 2) $\langle \text{nationality: han} \rangle \wedge \langle \text{department: economics} \rangle$

$\Rightarrow \langle \text{credit: 1} \rangle \wedge \langle \text{department: human-resource} \rangle$

(Rule 3) $\langle \text{politics: league-member} \rangle \wedge \langle \text{nationality: han} \rangle \wedge \langle \text{department: economics} \rangle \wedge \langle \text{length-of-schooling: 4} \rangle$
 $\Rightarrow \langle \text{credit: 1} \rangle \wedge \langle \text{department: Human-resource} \rangle$
(Rule 4) $\langle \text{birth-year: 1993} \rangle \wedge \langle \text{nationality: han} \rangle \wedge \langle \text{length-of-schooling: 4} \rangle \wedge \langle \text{grade: 2011} \rangle$
 $\Rightarrow \langle \text{credit: 1} \rangle \wedge \langle \text{department: human-resource} \rangle$

All rules are quite meaningful, and they might be employed for course recommendation directly. Rule 1 indicates that students in the economics like courses offered by the human-resource department. We observe that Rule 3 is finer than Rule 2, which is in turn finer than Rule 1. It happens that all three rules hold under the given setting. Rule 4 is not comparable with other three rules in terms of granulation.

5.2.2. The performance of dedicated rule checking approaches

We study the performance of the sandwich algorithm for different subtypes. We focus on Step 3 of the algorithm since it is most time consuming than Steps 1 and 2 for large datasets, and it is different for subtypes. The algorithm chooses the appropriate subtype according to mc and tc settings, as indicated in Section 4.2. We deliberately set mc and/or tc to 0.95 such that different subtypes are chosen, while the rule set is the same as the case of 1.

The results are listed in Table 5, where *basic operation* refers to comparison, addition, etc. Here we observe that the dedicated approaches for three subtypes are significantly faster than the one for the general case. For example, when $ms = mt = 0.01$, approaches for complete match subtype, left-hand side partial match subtype, and right-hand side partial match subtype are 866, 128, and 174 times faster than the general partial match subtype. Generally, the speed of algorithms for three subtypes are 2-3 orders of magnitudes faster than the one for the general case.

5.2.3. The performance of different algorithms

We compare the sandwich algorithm, the forward algorithm and the backward algorithm for the complete match subtype. Only the number of basic operations are compared, as depicted in Figure 2. Here we set $ms = mt$ and let them range from 0.007 to 0.01. We observe that the forward algorithm generally perform the best. It is about one time faster than the sandwich algorithm.

Note that the speed up is not as significant as indicated by Equations (41), (44) and (45). Nor does the backward algorithm outperform the forward algorithm when $|U| < |V|$. One important reason is that rule checking terminates once certain conditions are met, therefore introducing much uncertainty to the run time. Consequently, the time complexities are for reference only, and the run time depends more on the characteristics of data. This might be a common phenomenon for data mining algorithms.

6. Conclusions and further works

In this paper, we have proposed granular association rules to reveal many-to-many relationships in relational databases. They have wide applications such

Table 5: Run time of Step 3 for different settings

ms	mt	$ SC(ms) $	$ TC(mt) $	mc	tc	basic operations	run time (ms)
0.06	0.06	670	45	1	1	30,214	0
				0.95	1	2,479,057	31
				1	0.95	904,500	0
				0.95	0.95	1,462,868,100	8,594
0.05	0.05	817	55	1	1	44,999	0
				0.95	1	3,251,891	31
				1	0.95	1,168,310	0
				0.95	0.95	1,662,924,120	10,375
0.04	0.04	1,041	91	1	1	95,371	0
				0.95	1	5,860,127	31
				1	0.95	1,722,855	16
				0.95	0.95	2,085,064,990	13,156
0.03	0.03	2,268	113	1	1	259,596	0
				0.95	1	9,630,946	63
				1	0.95	4,003,020	31
				0.95	0.95	2,925,501,620	17,547
0.02	0.02	5,385	187	1	1	1,020,287	16
				0.95	1	23,211,961	156
				1	0.95	10,866,930	78
				0.95	0.95	4,831,951,668	29,594
0.01	0.01	18,160	275	1	1	5,067,570	63
				0.95	1	59,233,899	422
				1	0.95	39,845,600	312
				0.95	0.95	8,898,295,754	54,625

as collaborative recommendation [4] and collaborative filtering [15]. Four measures have been defined to evaluate the quality of these rules. Therefore the new type of rules are semantically richer than existing ones. We also proposed three algorithms for association rule mining, and compared algorithm efficiency through experimentation.

The following research topics deserve further investigation:

1. Different types of data for object description. In this work we considered only symbolic data for describing objects. It is necessary to consider numeric data, heterogenous data [18], interval valued [7] data and data with missing values [38]. There are some neighborhood systems concerning distance [18] or error ranges [30] to formalize these data. Respective approaches (see, e.g., [7, 18, 30]) can be also employed for these issues. Moreover, there might be test cost while obtaining data [29, 28]. Hence we should also consider cost data in certain applications.
2. Different granular association rule mining problems. In the problem def-

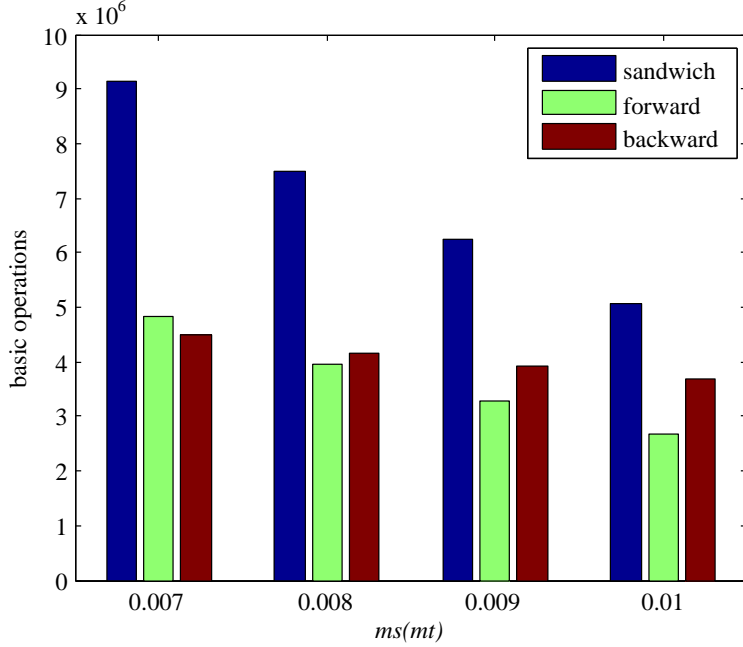


Figure 2: Basic operations of three algorithms

initiation of this paper, four thresholds are needed as the input. We may provide other means of parameter setting for non-expert users. For example, we may mine top- K interesting rules where K is easy to specify. We may need to remove redundant rules [32, 34] and common sense rules to avoid pattern explosion [37].

3. Efficient algorithms to these problems. As discussed in Section 4, the time complexities of proposed algorithms are rather high. For datasets with hundreds of thousands of objects, these algorithms may take too much time. Therefore we need to improve the speed of the algorithms dramatically through taking fully advantage of the Apriori property indicated in Section 4. Rough sets approach to association rule mining [31] may be also employed for this purpose. Moreover, since our algorithms are essentially exhaustive ones, it may be even necessary to design heuristic algorithms for large datasets. Consequently, we may design heuristic algorithms [37] to these problems.
4. Theoretical foundations of these problems and algorithms. The forward and the backward algorithms make use of concept approximation from the viewpoint of rough sets [33], especially the one for two universes [24, 39]. These two algorithms consider only complete matching rules, therefore the classical rough set model is employed. For the general case and two other subtypes, we may need variable precision rough sets [45] or decision

theoretical rough sets [20, 23, 26, 41]. There are at least two types of coverings induced by binary relations in this scenario. The first type of coverings is induced by binary relations. Given an element in one universe, the binary relation always induces a subset in other. In this way, from all elements in one universe, a cover of the other universe is induced. The second type of coverings is induced by granular association rules. Either side of a rule corresponds to a concept, which describes a covering block. Covering-based rough sets [27, 43, 44] are a natural approach for these issues.

To sum up, granular association rule mining is a challenging problem due to pattern explosion [37]. It may benefit from rough sets, especially variable precision rough sets [45] and covering-based rough sets [44]. Therefore this work has opened a new research trend concerning granular computing, association rule mining, and rough sets.

Acknowledgements

We would like to thank Mrs. Chunmei Zhou for her help in the data collection. This work is in part supported by National Science Foundation of China under Grant No. 61170128, the Natural Science Foundation of Fujian Province, China under Grant Nos. 2011J01374, 2012J01294, State key laboratory of management and control for complex systems open project under Grant No. 20110106.

References

- [1] F. Afrati, G. Das, A. Gionis, H. Mannila, T. Mielikäinen, P. Tsaparas, Mining chains of relations, in: *Data Mining: Foundations and Intelligent Paradigms*, vol. 24, Springer, 2012, pp. 217–246.
- [2] R. Agrawal, T. Imieliński, A. Swami, Mining association rules between sets of items in large databases, in: *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, 1993, pp. 207–216.
- [3] R. Agrawal, R. Srikant, Fast algorithms for mining association rules in large databases, in: *Proceedings of the 20th International Conference on Very Large Data Bases*, 1994, pp. 487–499.
- [4] M. Balabanović, Y. Shoham, Fab: content-based, collaborative recommendation, *Communication of ACM* 40 (3) (1997) 66–72.
- [5] A. Bargiela, W. Pedrycz, *Granular Computing: An Introduction*, Kluwer Academic Publishers, Boston, 2002.
- [6] H. Blockeel, L. D. Raedt, Top-down induction of first order logical decision trees, *Artificial Intelligence* 101 (1998) 285–297.

- [7] J. Dai, W. Wang, Q. Xua, H. Tian, Uncertainty measurement for interval-valued decision systems based on extended conditional entropy, *Knowledge-Based Systems* 27 (2012) 443–450.
- [8] L. Dehaspe, H. Toivonen, Discovery of frequent datalog patterns, *Expert Systems with Applications* 3 (1) (1999) 7–36.
- [9] L. Dehaspe, H. Toivonen, R. D. King, Finding frequent substructures in chemical compounds, in: 4th International Conference on Knowledge Discovery and Data Mining, 1998, pp. 30–36.
- [10] S. Džeroski, Multi-relational data mining: An introduction, in: *SIGKDD Explorations*, vol. 5, 2003, pp. 1–16.
- [11] S. Džeroski, N. Lavrac (eds.), *Relational data mining*, Springer, 2001.
- [12] W. Emde, D. Wettschereck, Relational instance-based learning, in: *Proceedings of the 13th International Conference on Machine Learning*, 1996, pp. 122–130.
- [13] B. Goethals, W. L. Page, M. Mampaey, Mining interesting sets and rules in relational databases, in: *Proceedings of the 2010 ACM Symposium on Applied Computing*, 2010, pp. 997–1001.
- [14] B. Goethals, W. L. Page, H. Mannila, Mining association rules of simple conjunctive queries, in: *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 2008, pp. 96–107.
- [15] D. Goldberg, D. Nichols, B. M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, *Communications of the ACM* 35 (1992) 61–70.
- [16] J. Han, Y. Fu, Discovery of multi-level association rules from large databases, in: *Proceedings of the International Conference on Very Large Databases*, 1995, pp. 420–431.
- [17] J. Han, J. Pei, Y. Yin, Mining frequent patterns without candidate generation, in: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, ACM, 2000, pp. 1–12.
- [18] Q. H. Hu, D. R. Yu, Z. Xie, Numerical attribute reduction based on neighborhood granulation and rough approximation (in chinese), *Journal of Software* 19 (3) (2008) 640–649.
- [19] V. C. Jensen, N. Soparkar, Frequent itemset counting across multiple tables, in: *Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Application*, vol. 1805 of LNCS, 2000, pp. 49–61.

- [20] X. Y. Jia, W. H. Liao, Z. M. Tang, L. Shang, Minimum cost attribute reduction in decision-theoretic rough set models, *Information Sciences*, doi: j.ins.2012.07.010.
- [21] Y. Kavurucu, P. Senkul, I. Toroslu, ILP-based concept discovery in multi-relational data mining, *Expert Systems with Applications* 36 (2009) 11418–11428.
- [22] S. Kramer, Structural regression trees, in: *Proceedings of the 13th National Conference on Artificial Intelligence*, 1996, pp. 812–819.
- [23] H. X. Li, X. Z. Zhou, Risk decision making based on decision-theoretic rough set: a three-way view decision model, *International Journal of Computational Intelligence Systems* 4 (1) (2011) 1–11.
- [24] T. J. Li, W. X. Zhang, Rough fuzzy approximations on two universes of discourse, *Information Sciences* 178 (3) (2008) 892–906.
- [25] T. Y. Lin, Granular computing on binary relations i: Data mining and neighborhood systems, in: *Rough Sets in Knowledge Discovery*, 1998, pp. 107–121.
- [26] D. Liu, T. R. Li, P. Hu, H. X. Li, Multiple-category classification with decision-theoretic Rough sets, in: *Proceedings of Rough Sets and Knowledge Technology*, vol. 6401 of LNAI, 2010, pp. 703–710.
- [27] G. Liu, W. Zhu, The algebraic structures of generalized rough set theory, *Information Sciences* 178 (21) (2008) 4105–4113.
- [28] F. Min, H. P. He, Y. H. Qian, W. Zhu, Test-cost-sensitive attribute reduction, *Information Sciences* 181 (2011) 4928–4942.
- [29] F. Min, Q. H. Liu, A hierarchical model for test-cost-sensitive decision systems, *Information Sciences* 179 (2009) 2442–2452.
- [30] F. Min, W. Zhu, Attribute reduction of data with error ranges and test costs, *Information Sciences* 211 (2012) 48–67.
- [31] H. S. Nguyen, D. Ślęzak, Approximate reducts and association rules: correspondence and complexity results, in: *Proceedings of Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, vol. 1711 of LNAI, 1999, pp. 137–145.
- [32] N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal, Discovering frequent closed itemsets for association rules, in: *Proceedings of the 7th International Conference on Database Theory*, 1999, pp. 398–416.
- [33] Z. Pawlak, Rough sets, *International Journal of Computer and Information Sciences* 11 (1982) 341–356.

- [34] J. Pei, J. Han, R. Mao, CLOSET: An efficient algorithm for mining frequent closed itemsets, in: ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 2000, pp. 21–30.
- [35] A. Skowron, J. Stepaniuk, Approximation of relations, in: W. Ziarko (ed.), Proceedings of Rough Sets, Fuzzy Sets and Knowledge Discovery, 1994, pp. 161–166.
- [36] R. Srikant, R. Agrawal, Mining quantitative association rules in large relational tables, SIGMOD Rec. 25 (2) (1996) 1–12.
- [37] J. Vreeken, M. Leeuwen, A. Siebes, Krimp: mining itemsets that compress, Data Mining and Knowledge Discovery 23 (1) (2011) 169–214.
- [38] Q. Yang, X. Wu, 10 challenging problems in data mining research, International Journal of Information Technology and Decision Making 5 (4) (2006) 597–604.
- [39] Y. Y. Yao, Two views of the theory of rough sets in finite universes, International Journal of Approximate Reasoning 15 (1996) 291–317.
- [40] Y. Y. Yao, Granular computing: basic issues and possible solutions, in: Proceedings of the 5th Joint Conference on Information Sciences, vol. 1, 2000, pp. 186–189.
- [41] Y. Y. Yao, S. Wong, A decision theoretic framework for approximating concepts, International Journal of Man-machine Studies 37 (1992) 793–809.
- [42] L. Zadeh, Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, Fuzzy Sets and Systems 19 (1997) 111–127.
- [43] W. Zhu, Relationship among basic concepts in covering-based rough sets, Information Sciences 17 (14) (2009) 2478–2486.
- [44] W. Zhu, F. Wang, Reduction and axiomization of covering generalized rough sets, Information Sciences 152 (1) (2003) 217–230.
- [45] W. Ziarko, Variable precision rough set model, Journal of Computer and System Sciences 46 (1) (1993) 39–59.